

**IN THE CLAIMS:**

Please cancel claims 18-25, 42-49 and 52-53. The remaining claims are presented as follows:

1. (Previously Presented) A method of managing execution of query operations in a data processing system, comprising:  
issuing, by a requesting entity, a request to perform a composite query operation defined by at least an initial query operation and a subsequent query operation to be executed against a data repository of the data processing system;  
executing the initial query operation;  
determining an operation status of the initial query operation; and  
managing execution of the subsequent query operation on the basis of the determined operation status.
2. (Original) The method of claim 1, wherein the determining and managing are performed by a composite query operations manager.
3. (Original) The method of claim 1, wherein the requesting entity is an application and wherein the determining and managing are performed by a composite query operations manager.
4. (Original) The method of claim 1, wherein the initial and the subsequent query operation are SQL statements.
5. (Original) The method of claim 1, wherein determining an operation status of the initial query operation comprises determining a number of items affected by the initial query operation.

6. (Original) The method of claim 1, wherein determining an operation status of the initial query operation comprises determining whether the initial query operation completed successfully.

7. (Original) The method of claim 1, wherein determining an operation status of the initial query operation comprises determining, on the basis of a return code received upon completion of the initial query operation, whether the initial query operation completed successfully.

8. (Original) The method of claim 1, wherein managing execution of the subsequent query operation comprises:

executing the subsequent query operation only if the initial query operation did not complete successfully.

9. (Previously Presented) A method of managing execution of query operations in a data processing system, comprising:

issuing, by a requesting entity, a request to perform a composite query operation defined by at least an initial query operation and a plurality of subsequent query operations to be executed against a data repository of the data processing system;

providing selection logic defining a next query operation of the composite query operation to be executed;

providing a plurality of failure conditions for determining when a failure of the composite query operation occurs; and

managing, using a composite query operations manager, execution of the initial query operation and the plurality of subsequent query operations on the basis of the selection logic and the plurality of failure conditions.

10. (Original) The method of claim 9, wherein at least one failure condition of the plurality of failure conditions indicates the initial query operation and an operation status of the initial query operation which indicates a failure of the composite query operation.

11. (Original) The method of claim 10, wherein each other failure condition of the plurality of failure conditions indicates a series of the initial query operation and at least one of the plurality of subsequent query operations, and an operation status of the at least one of the plurality of subsequent query operations which indicates a failure of the composite query operation.

12. (Original) The method of claim 9, wherein managing execution of the initial query operation and the plurality of subsequent query operations comprises:

- a) executing the initial query operation;
- b) determining an operation status of the initial query operation;
- c) determining, on the basis of the operation status and the plurality of failure conditions, whether failure of the composite query operation occurred; and
- d) if no failure of the composite query operation occurred:
  - i) determining the next operation to be executed from the plurality of subsequent query operations using the selection logic;
  - ii) executing the next query operation;
  - iii) determining an operation status of the next query operation;
  - iv) determining, on the basis of the operation status and the plurality of failure conditions, whether failure of the composite query operation occurred; and
  - v) repeating step d) for at least one other of the plurality of subsequent query operations.

13. (Original) The method of claim 12, wherein step c) and iv) comprise, if failure occurred:

- completing execution of the composite query operation; and
- returning a failure code as completion status of the composite query operation indicating a failure condition from the plurality of failure conditions.

14. (Original) The method of claim 12, wherein step b) and step iii) comprise determining whether the respective query operation completed successfully.
15. (Original) The method of claim 12, wherein step b) and iii) comprise determining, on the basis of a return code received upon completion of the respective query operation, whether the respective query operation completed successfully.
16. (Original) The method of claim 12, wherein step b) and step iii) comprise determining whether the respective query operation completed successfully and, if the respective query operation completed successfully:  
completing execution of the composite query operation.
17. (Original) The method of claim 9, wherein the initial and each subsequent query operation is an SQL statement.
18. (Cancelled) A method of managing execution of query operations in a data processing system, comprising:  
providing a composite query operation to be executed against a data repository of the data processing system;  
providing at least two implementation schemas for the composite query operation, each defining a different order of execution of at least two different query operations required to perform the composite query operation;  
executing the composite query operation according to a first implementation schema of the implementation schemas;  
determining, upon completion of the execution of the composite query operation, a completion status of the composite query operation; and  
managing a subsequent execution of the composite query operation on the basis of the determined completion status.

19. (Cancelled) The method of claim 18, wherein the completion status indicates a number of query operations executed as part of the composite query operation.

20. (Cancelled) The method of claim 18, wherein the completion status indicates a number of query operations executed as part of the composite query operation; and wherein the managing comprises:

executing the composite query operation according to a second implementation schema of the at least two implementation schemas, if the number of query operations exceeds a predetermined threshold.

21. (Cancelled) The method of claim 18, wherein the completion status indicates a number of query operations executed as part of the composite query operation; and wherein the managing comprises:

initializing a failure count;

repeatedly executing the composite query operation in response to subsequent requests for execution of the composite query operation according to the first implementation schema; and

upon each execution of the composite query operation according to the first implementation schema:

incrementing the failure count if the number of query operations indicated by the completion status exceeds a first predetermined threshold; and

if the failure count exceeds a second predetermined threshold:

resetting the failure count; and

executing the composite query operation according to a second implementation schema of the implementation schemas.

22. (Cancelled) The method of claim 18, wherein the completion status indicates a number of query operations executed as part of the composite query operation; and wherein the managing comprises:

initializing an accumulated number of query operations;

repeatedly executing the composite query operation in response to subsequent requests for execution of the composite query operation according to the first implementation schema; and

upon each execution of the composite query operation according to the first implementation schema:

summing the number of query operations indicated by the completion status to the accumulated number of query operations; and

if the accumulated number of query operations exceeds a predetermined threshold:

resetting the accumulated number of query operations; and

executing the composite query operation according to a second implementation schema of the implementation schemas.

23. (Cancelled) The method of claim 18, wherein the first and second query operations are SQL statements.

24. (Cancelled) A method of managing execution of query operations in a data processing system, comprising:

a) providing a composite query operation defining a first and a second query operation to be executed against a data repository of the data processing system;

b) providing a first and a second implementation schema for the composite query operation, each defining a different order of execution of the first and the second query operation;

c) selecting an implementation schema of the first and second implementations schemas;

d) initializing a failure count;

e) repeatedly executing the composite query operation according to the selected implementation schema in response to subsequent requests for execution of the composite query operation according to the selected implementation schema; and

f) upon each execution of the composite query operation according to the selected implementation schema:

f1) determining a completion status of the composite query operation indicating a number of query operations executed as part of the composite query operation;

f2) incrementing the failure count if the number of query operations indicated by the completion status exceeds a first predetermined threshold;

f3) determining whether the failure count exceeds a second predetermined threshold; and

f4) if the failure count exceeds the second predetermined threshold:

i) resetting the failure count;

ii) selecting the other implementation schema; and

iii) returning to step e).

25. (Cancelled) The method of claim 24, wherein the first and second query operations are SQL statements.

26. (Previously Presented) A computer readable storage medium containing a program which, when executed, performs a process of managing execution of query operations in a data processing system, the process comprising:

receiving, from a requesting entity, a request to perform a composite query operation defined by at least an initial query operation and a subsequent query operation to be executed against a data repository of the data processing system;

executing the initial query operation;

determining an operation status of the initial query operation; and

managing execution of the subsequent query operation on the basis of the determined operation status.

27. (Previously Presented) The computer readable storage medium of claim 26, wherein the determining and managing are performed by a composite query operations manager.

28. (Previously Presented) The computer readable storage medium of claim 26, wherein the initial and the subsequent query operation are SQL statements.

29. (Previously Presented) The computer readable storage medium of claim 26, wherein determining an operation status of the initial query operation comprises determining a number of items affected by the initial query operation.

30. (Previously Presented) The computer readable storage medium of claim 26, wherein determining an operation status of the initial query operation comprises determining whether the initial query operation completed successfully.

31. (Previously Presented) The computer readable storage medium of claim 26, wherein determining an operation status of the initial query operation comprises determining, on the basis of a return code received upon completion of the initial query operation, whether the initial query operation completed successfully.

32. (Previously Presented) The computer readable storage medium of claim 26, wherein managing execution of the subsequent query operation comprises:  
executing the subsequent query operation only if the initial query operation did not complete successfully.

33. (Previously Presented) A computer readable storage medium containing a program which, when executed, performs a process of managing execution of query operations in a data processing system, the process comprising:

receiving, from a requesting entity, a request to perform a composite query operation defined by at least an initial query operation and a plurality of subsequent



query operations to be executed against a data repository of the data processing system;

retrieving selection logic defining a next query operation of the composite query operation to be executed;

retrieving a plurality of failure conditions for determining when a failure of the composite query operation occurs; and

managing, using a composite query operations manager, execution of the initial query operation and the plurality of subsequent query operations on the basis of the selection logic and the plurality of failure conditions.

34. (Previously Presented) The computer readable storage medium of claim 33, wherein at least one failure condition of the plurality of failure conditions indicates the initial query operation and an operation status of the initial query operation which indicates a failure of the composite query operation.

35. (Previously Presented) The computer readable storage medium of claim 34, wherein each other failure condition of the plurality of failure conditions indicates a series of the initial query operation and at least one of the plurality of subsequent query operations, and an operation status of the at least one of the plurality of subsequent query operations which indicates a failure of the composite query operation.

36. (Previously Presented) The computer readable storage medium of claim 33, wherein managing execution of the initial query operation and the plurality of subsequent query operations comprises:

- a) executing the initial query operation;
- b) determining an operation status of the initial query operation;
- c) determining, on the basis of the operation status and the plurality of failure conditions, whether failure of the composite query operation occurred; and
- d) if no failure of the composite query operation occurred:

- i) determining the next operation to be executed from the plurality of subsequent query operations using the selection logic;
- ii) executing the next query operation;
- iii) determining an operation status of the next query operation;
- iv) determining, on the basis of the operation status and the plurality of failure conditions, whether failure of the composite query operation occurred; and
- v) repeating step d) for at least one other of the plurality of subsequent query operations.

37. (Previously Presented) The computer readable storage medium of claim 36, wherein step c) and iv) comprise, if failure occurred:

- completing execution of the composite query operation; and
- returning a failure code as completion status of the composite query operation indicating a failure condition from the plurality of failure conditions.

38. (Previously Presented) The computer readable storage medium of claim 36, wherein step b) and step iii) comprise determining whether the respective query operation completed successfully.

39. (Previously Presented) The computer readable storage medium of claim 36, wherein step b) and iii) comprise determining, on the basis of a return code received upon completion of the respective query operation, whether the respective query operation completed successfully.

40. (Previously Presented) The computer readable storage medium of claim 36, wherein step b) and step iii) comprise determining whether the respective query operation completed successfully and, if the respective query operation completed successfully:

completing execution of the composite query operation.

41. (Previously Presented) The computer readable storage medium of claim 33, wherein the initial and each subsequent query operation is an SQL statement.

42. (Cancelled) A computer readable storage medium containing a program which, when executed, performs a process of managing execution of query operations in a data processing system, the process comprising:

receiving a request to perform a composite query operation against a data repository of the data processing system;

providing at least two implementation schemas for the composite query operation, each defining a different order of execution of at least two different query operations required to perform the composite query operation;

executing the composite query operation according to a first implementation schema of the implementation schemas;

determining, upon completion of the execution of the composite query operation, a completion status of the composite query operation; and

managing a subsequent execution of the composite query operation on the basis of the determined completion status.

43. (Cancelled) The computer readable storage medium of claim 42, wherein the completion status indicates a number of query operations executed as part of the composite query operation.

44. (Cancelled) The computer readable storage medium of claim 42, wherein the completion status indicates a number of query operations executed as part of the composite query operation; and wherein the managing comprises:

executing the composite query operation according to a second implementation schema of the at least two implementation schemas, if the number of query operations exceeds a predetermined threshold.

45. (Cancelled) The computer readable storage medium of claim 42, wherein the completion status indicates a number of query operations executed as part of the composite query operation; and wherein the managing comprises:

initializing a failure count;

repeatedly executing the composite query operation in response to subsequent requests for execution of the composite query operation according to the first implementation schema; and

upon each execution of the composite query operation according to the first implementation schema:

incrementing the failure count if the number of query operations indicated by the completion status exceeds a first predetermined threshold; and

if the failure count exceeds a second predetermined threshold:

resetting the failure count; and

executing the composite query operation according to a second implementation schema of the implementation schemas.

46. (Cancelled) The computer readable storage medium of claim 42, wherein the completion status indicates a number of query operations executed as part of the composite query operation; and wherein the managing comprises:

initializing an accumulated number of query operations;

repeatedly executing the composite query operation in response to subsequent requests for execution of the composite query operation according to the first implementation schema; and

upon each execution of the composite query operation according to the first implementation schema:

summing the number of query operations indicated by the completion status to the accumulated number of query operations; and  
if the accumulated number of query operations exceeds a predetermined threshold:

resetting the accumulated number of query operations; and  
executing the composite query operation according to a second implementation schema of the implementation schemas.

47. (Cancelled) The computer readable storage medium of claim 42, wherein the first and second query operations are SQL statements.

48. (Cancelled) A computer readable storage medium containing a program which, when executed, performs a process of managing execution of query operations in a data processing system, the process comprising:

a) receiving a request to perform a composite query operation defined by at least a first and a second query operation to be executed against a data repository of the data processing system;

b) retrieving a first and a second implementation schema for the composite query operation, each defining a different order of execution of the first and the second query operation;

c) selecting an implementation schema of the first and second implementations schemas;

d) initializing a failure count;

e) repeatedly executing the composite query operation according to the selected implementation schema in response to subsequent requests for execution of the composite query operation according to the selected implementation schema; and

f) upon each execution of the composite query operation according to the selected implementation schema:

- f1) determining a completion status of the composite query operation indicating a number of query operations executed as part of the composite query operation;
  - f2) incrementing the failure count if the number of query operations indicated by the completion status exceeds a first predetermined threshold;
  - f3) determining whether the failure count exceeds a second predetermined threshold; and
  - f4) if the failure count exceeds the second predetermined threshold:
    - i) resetting the failure count;
    - ii) selecting the other implementation schema; and
    - iii) returning to step e).
49. (Cancelled) The computer readable storage medium of claim 48, wherein the first and second query operations are SQL statements.
50. (Previously Presented) A data processing system comprising:  
a data repository; and  
a composite query operations manager residing in memory for managing execution of query operations in the data processing system, the composite query operations manager being configured for:  
receiving a request to perform a composite query operation defined by at least an initial query operation and a subsequent query operation to be executed against the data repository;  
executing the initial query operation;  
determining an operation status of the initial query operation; and  
managing execution of the subsequent query operation on the basis of the determined operation status.
51. (Previously Presented) A data processing system comprising:  
a data repository; and

a composite query operations manager residing in memory for managing execution of query operations in the data processing system, the composite query operations manager being configured for:

- receiving a request to perform a composite query operation defined by at least an initial query operation and a plurality of subsequent query operations to be executed against the data repository;

- retrieving selection logic defining a next query operation of the composite query operation to be executed;

- retrieving a plurality of failure conditions for determining when a failure of the composite query operation occurs; and

- managing execution of the initial query operation and the plurality of subsequent query operations on the basis of the selection logic and the plurality of failure conditions.

52. (Cancelled) A data processing system comprising:

- a data repository; and

- a composite query operations manager residing in memory for managing execution of query operations in the data processing system, the composite query operations manager being configured for:

- receiving a request to perform a composite query operation against a data repository of the data processing system;

- providing at least two implementation schemas for the composite query operation, each defining a different order of execution of at least two different query operations required to perform the composite query operation;

- executing the composite query operation according to a first implementation schema of the implementation schemas;

- determining, upon completion of the execution of the composite query operation, a completion status of the composite query operation; and

- managing a subsequent execution of the composite query operation on the basis of the determined completion status.

53. (Cancelled) A data processing system comprising:

a data repository; and

a composite query operations manager residing in memory for managing execution of query operations in the data processing system, the composite query operations manager being configured for:

a) receiving a request to perform a composite query operation defined by at least a first and a second query operation to be executed against the data repository;

b) retrieving a first and a second implementation schema for the composite query operation, each defining a different order of execution of the first and the second query operation;

c) selecting an implementation schema of the first and second implementations schemas;

d) initializing a failure count;

e) repeatedly executing the composite query operation according to the selected implementation schema in response to subsequent requests for execution of the composite query operation according to the first implementation schema; and

f) upon each execution of the composite query operation according to the selected implementation schema:

f1) determining a completion status of the composite query operation indicating a number of query operations executed as part of the composite query operation;

f2) incrementing the failure count if the number of query operations indicated by the completion status exceeds a first predetermined threshold;

f3) determining whether the failure count exceeds a second predetermined threshold; and



f4) if the failure count exceeds the second predetermined threshold:

- i) resetting the failure count;
- ii) selecting the other implementation schema; and
- iii) returning to step e).

54. (Cancelled) A data structure residing in memory, comprising:
- at least one composite query operation component including:
    - a plurality of query operations adapted for defining a composite query operation to be executed against a data repository of a data processing system;
    - selection logic defining a next query operation of the composite query operation to be executed; and
    - a plurality of failure conditions for determining when a failure of the composite query operation occurs; and
  - at least two implementation schemas for the composite query operation, each defining a different order of execution of at least two different query operations required to perform the composite query operation.